

Maximo JSON API Reference

V1 - Updated: 12/17/2015

Contents

- Introduction 2
- Query Parameters 2
- HTTP Headers 4
- System Properties 5
- HTTP codes 6

Introduction

This document covers Reference information related to using the JSON API. You should review the Maximo JSON API Overview document prior to using this document.

Query Parameters

Note: all Booleans support values of 1/0 or true/false (1/0 are used in the table below)

Name	Allowed	Default	Description
lean	1/0	0	when set to 1 (true), this will instruct the api to work with the "lean" JSON format, not the namespaced JSON format that is supported by default.
oslc.where			Filter clause applied to a collection query following the OSLC query syntax. Allows for selecting a subset of resources within a collection.
oslc.select			Select clause applied to a collection query following the OSLC query syntax. Allows for selecting a partial view (limited set of fields) of the resource data in the collection.
oslc.properties			Properties clause applied to a resource member query following the OSLC query syntax. Allows for selecting a partial view (limited set of fields) of the member resource
oslc.orderBy			Order By clause applied to a collection query following the OSLC query syntax. Allows for sorting of the resource data in the collection.
opmodeor	1/0	0	When set to 1 (true), this will direct the oslc.where clause to use OR rather than AND when the clause contains multiple conditions
collectioncount	1/0	0	When set to 1 (true), it will direct the API to include the total count of records selected by the query and is available in the response as responseInfo.totalCount. This parameter overrides the default setting provided by the maximo property mxe.oslc.collectioncount.
oslc.pageSize			Directs the server to page the query result with the specified page size, such as 10 records per page.
oslc.searchTerms			Record wide text search for values specified in this query parameter. Example oslc.searchTerms="pump","broken". Values are evaluated as an 'OR' condition the field must have a value of either pump or broken, but does not have to have both. The values provided will

			be used for searching against the fields identified in the searchAttributes parameter. Both parameters must be provided.
searchAttributes			Identifies the resource field(s) to be searched using the values provided in the osc.searchTerms. Example searchAttributes=description,status. Both parameters must be provided.
stablepaging	1	0	if set to true, will instruct the server to page the resource from an in session MboSet.

HTTP Headers

Name	Allowed	Default	Description
x-method-override	PATCH/ DELETE/ BULK		Due to constraints from browser the HTTP POST method is re-used for creates and updates and deletes. This request header is used to convey the specifics of the POST request. PATCH is used for update. DELETE is used for deleting the resource. BULK is used for supporting bulk operations. If this header is not present, the POST request is treated as a create request.
x-allow-events	1/0	1	For a create/update/delete, if the request wants to suppress the object events, this request header needs to be set to 0. By default all events are fired unless programmatically stopped.
transactionid			This is used to prevent duplicate transactions when sent multiple times by a client. If this header is provided with a unique ID then the value is compared to a list of previously processed IDs. If a match exists then it is considered as a duplicate transaction and an HTTP conflict error (409) is generated.
x-document-meta	Value as per the "URLTYPE" domain value and a valid doctype.		A sample header value for OOTB Maximo is FILE/Attachments. Do make sure that the attachments setup is completed on the Maximo server side before attempting to use this api feature.
x-document-description			the description of the attachment.
x-document-encdescription			There is a limitation in HTTP which prevents non-ISO-8859-1 characters in any request header value. To set descriptions in other character sets,base64 encode this value and then set the value in the x-document-encdescription header.
slug			the name of the attached document file. An example pump.jpeg.
content-type			the mime-type of the request document.
patchtype	MERGE		This request header with the value of MERGE will instruct the server to treat a PATCH request (POST with x-method-override as PATCH) processor to merge the child objects (array of child objects) in the request with the corresponding child elements in the server. If this header is not present, the server will delete the child objects that are missing from the request.

System Properties

Below are system properties that impact the JSON API

Name	Default	Description
mxe.oslc.collectioncount	0	When true (1) provides a total count of resources in the Collection response.
mxe.oslc.webappurl	../localhost/..	Set with Maximo server's Host:Port
mxe.oslc.osname.fetchlimit	n/a	Property not provided out of box. Can be used as a safety net to prevent queries from returning large amounts of resources in a single request. Example: to limit the number of asset resources to 10,000 create a property with name mxe.oslc.mxasset.fetchlimit with a value of 10000
mxe.oslc.idleexpiry	300	Amount of time (in seconds) the resource data remains in memory when Stable Paging is used.
mxe.oslc.isosaresource	1	Provided to support backwards compatibility with the OSLC API. When set to True (1), this allows object structures to be used as JSON resources without the need to define a corresponding OSLC Resource.

HTTP codes

OSLC uses standard HTTP response codes as error messages. For example, the HTTP 404 response is normally returned when a web page is not found, and in OSLC the 404 response code is returned when the resource cannot be found.

The following HTTP response codes are used with the JSON API

HTTP code	OSLC explanation
200	Success
201	Success. The response contains a link.
204	Resource successfully updated. There is no response entity.
400	Error handling request. This error might be due to the request content or URI. For example, there might be a business logic validation error on the server side.
401	Authentication failure.
403	Forbidden. The user password expired.
404	Resource cannot be found or an invalid resource type was provided.
405	HTTP method cannot be used for the resource.
406	Requested representation is not supported.
409	Conflict - Transaction ID previously processed
410	Stable resource page expired.

412	Resource on the client side is stale and must be refreshed from the server. The conditional update failed because the resource was updated by another user or process.
500	All other server errors.